1 Hand-written Part

P1

$1 - yw^Tx \geq 0 \iff yw^Tx \leq 1$

$$\begin{cases} y_nw^Tx_n \leq 1 \rightarrow \max(1-y_nw^Tx_n, 0) = 1 - y_nw^Tx_n \rightarrow \nabla E_n = \nabla(1-y_nw^Tx_n)^2 = -2y_nx_n(1-y_nw^Tx_n) \\ \qquad = -2y_nx_n + 2x_nw^Tx_n \\ y_nw^Tx_n \geq 1 \rightarrow \max(1-y_nw^Tx_n, 0) = 0 \rightarrow \nabla E_n = \nabla 0^2 = 0 \end{cases}$$

$$\Rightarrow \nabla E_{in}(w) = \nabla\left(\frac{1}{N}\sum_{n=1}^{N} \max(1-y_nw^Tx_n, 0)^2\right) = \frac{1}{N}\sum_{n=1}^{N} [\![y_nw^Ty_n \leq 1]\!] (-2y_nx_n + 2x_nw^Tx_n)$$

P2

$$\ln\left(\prod_{n=1}^{N} p_u(x_n)\right) = \sum_{n=1}^{N} \ln p_u(x_n) = \sum_{n=1}^{N} \ln \frac{e^{(-\frac{1}{2}(x-\mu)^T I^{-1}(x-\mu))}}{\sqrt{(2\pi)^d |I|}}$$

$$= \sum_{n=1}^{N} -\frac{1}{2}(x-\mu)^T(x-\mu) - N\ln\sqrt{(2\pi)^d|I|}$$

$$= \sum_{n=1}^{N} -\frac{1}{2}\|x-\mu\|^2 - N\ln\sqrt{(2\pi)^d|I|}$$

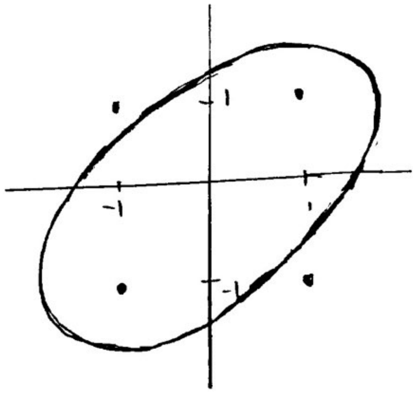
$$\max \prod_{n=1}^{N} p_u(x_n) \iff \max \ln\left(\prod_{n=1}^{N} p_u(x_n)\right) \iff \max \sum_{n=1}^{N} -\frac{1}{2}\|x_n-\mu\|^2 \iff \min \sum_{n=1}^{N}\|x_n-\mu\|^2$$

$$\iff \nabla_\mu\left(\sum_{n=1}^{N}\|x_n-\mu\|^2\right) = 0 \iff \sum_{n=1}^{N} -2(x_n-\mu) = 0 \iff \sum_{n=1}^{N} x_n = N\mu \iff \mu = \frac{1}{N}\sum_{n=1}^{N} x_n$$

P3

$\tilde{w} = (-2,0,0,1,-1,1) \rightarrow \tilde{w}^T\Phi_2((x_1,x_2)) = x_1^2 - x_1x_2 + x_2^2 - 2$

$$\left.\begin{array}{l} \text{sign}(\tilde{w}^Tz_1) = \text{sigh}(-1) = -1 = y_1 \\ \text{sign}(\tilde{w}^Tz_2) = \text{sigh}(+1) = +1 = y_2 \\ \text{sign}(\tilde{w}^Tz_3) = \text{sigh}(-1) = -1 = y_3 \\ \text{sign}(\tilde{w}^Tz_4) = \text{sigh}(+1) = +1 = y_4 \end{array}\right\} \quad \tilde{w} = (-2,0,0,1,-1,1) \text{ separates the data}$$



P4

Let $\overset{-}{\sum_n} f(n) = \sum_{n=1}^{N} \delta(g_t(x_n), y_n) f(n)$, $\overset{+}{\sum_n} f(n) = \sum_{n=1}^{N} (1-\delta(g_t(x_n), y_n)) f(n)$

$\Rightarrow \sum_{n=1}^{N} f(n) = \sum_{n=1}^{N} (\delta(g_t(x_n), y_n) + (1-\delta(g_t(x_n), y_n))) f(n) = \overset{-}{\sum_n} f(n) + \overset{+}{\sum_n} f(n)$

$$\varepsilon_{t+1} = \frac{\overset{-}{\sum_n} w_n^{t+1}}{\sum_{n=1}^{N} w_n^{t+1}} = \frac{\overset{-}{\sum_n} w_n^{t+1}}{\overset{-}{\sum_n} w_n^{t+1} + \overset{+}{\sum_n} w_n^{t+1}} = \frac{\overset{-}{\sum_n} d_t w_n^t}{\overset{-}{\sum_n} d_t w_n^t + \overset{+}{\sum_n} \frac{w_n^t}{d_t}} = \frac{d_t^2 \overset{-}{\sum_n} w_n^t}{d_t^2 \overset{-}{\sum_n} w_n^t + \overset{+}{\sum_n} w_n^t}$$

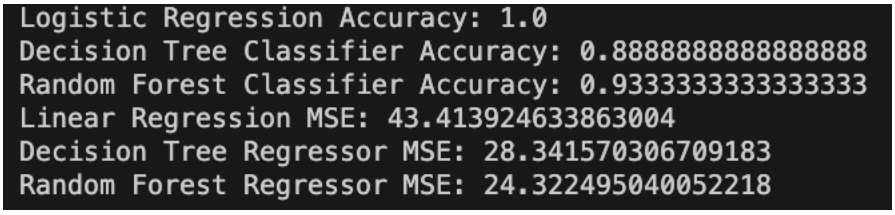
$$d_t^2 \overset{-}{\sum_n} w_n^t = \frac{1-\varepsilon_t}{\varepsilon_t} \times \overset{-}{\sum_n} w_n^t = \frac{\sum_{n=1}^{N} w_n^t - \sum_{n=1}^{N} w_n^t \varepsilon_t}{\sum_{n=1}^{N} w_n^t \varepsilon_t} \times \overset{-}{\sum_n} w_n^t = \frac{(\overset{+}{\sum_n} w_n^t + \overset{-}{\sum_n} w_n^t) - \overset{-}{\sum_n} w_n^t}{\overset{-}{\sum_n} w_n^t} \times \overset{-}{\sum_n} w_n^t$$

$$= \overset{+}{\sum_n} w_n^t$$

$$\varepsilon_{t+1} = \frac{d_t^2 \overset{-}{\sum_n} w_n^t}{d_t^2 \overset{-}{\sum_n} w_n^t + \overset{+}{\sum_n} w_n^t} = \frac{\overset{+}{\sum_n} w_n^t}{\overset{+}{\sum_n} w_n^t + \overset{+}{\sum_n} w_n^t} = \frac{1}{2}$$ ※

2 Programming Part

(a)



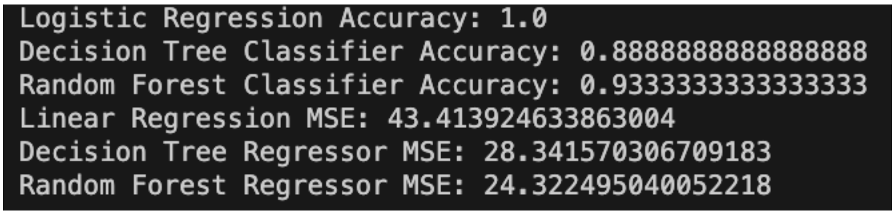
```
Logistic Regression Accuracy: 1.0
Decision Tree Classifier Accuracy: 0.8888888888888888
Random Forest Classifier Accuracy: 0.9333333333333333
Linear Regression MSE: 43.413924633863004
Decision Tree Regressor MSE: 28.341570306709183
Random Forest Regressor MSE: 24.322495040052218
```

For the classification task, the linear model achieves the highest performance, probably because the data is linear separable or the relationship between the features and the target variable is straightforward. Simpler models generally excel in such scenarios due to their lower variance.

For the regression task, the random forest model surpasses other models in performance. This is likely because it can manage non-linear and more intricate data effectively. Additionally, the bagging process helps reduce overfitting, giving it an edge over the decision tree model.

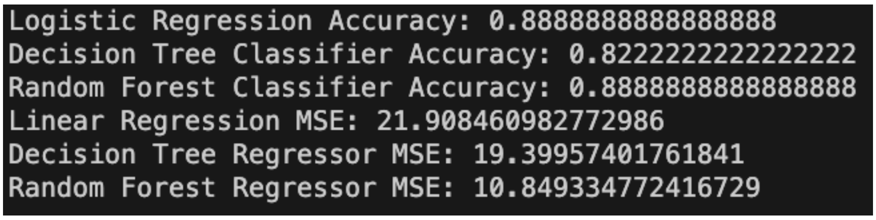
(b)

Normalization



```
Logistic Regression Accuracy: 1.0
Decision Tree Classifier Accuracy: 0.8888888888888888
Random Forest Classifier Accuracy: 0.9333333333333333
Linear Regression MSE: 43.413924633863004
Decision Tree Regressor MSE: 28.341570306709183
Random Forest Regressor MSE: 24.322495040052218
```

Standardization



```
Logistic Regression Accuracy: 0.8888888888888888
Decision Tree Classifier Accuracy: 0.8222222222222222
Random Forest Classifier Accuracy: 0.8888888888888888
Linear Regression MSE: 21.908460982772986
Decision Tree Regressor MSE: 19.39957401761841
Random Forest Regressor MSE: 10.849334772416729
```

Performance Impact on Logistic Regression:

- With **normalization**, logistic regression achieved an accuracy of 0.6667. This lower performance can be attributed to the sensitivity of logistic regression to feature scaling, as normalization might not have aligned the features effectively.
- With **standardization**, logistic regression's accuracy increased to 0.8889. This is likely because standardization better aligns the feature scales and centers the data, making it more suitable for the underlying assumptions of logistic regression.

Rationale and Comparison:

- **Rationale for Normalization:** Useful for datasets with varying scales but does not address the distribution of data. This can lead to suboptimal performance for algorithms sensitive to feature scales.
- **Rationale for Standardization:** Ensures that features contribute equally to the model by transforming them to a common scale. This is especially important for models like logistic regression that rely on the assumption of similarly scaled features.

Advantages and Disadvantages in Context:

- **Normalization:**
    - **Advantages:** Simple and effective for certain datasets.
    - **Disadvantages:** Sensitive to outliers and might not always align features appropriately for all models.
- **Standardization:**
    - **Advantages:** More robust and generally improves the performance of models sensitive to feature scaling.
    - **Disadvantages:** Assumes a normal distribution, which might not always be the case.

(c)

In experiments of linear regression models, it has been observed that a higher learning rate can accelerate the convergence towards the optimal solution. However, setting the learning rate too high can lead to the opposite effect, causing the model to diverge and move away from the desired solution. On the other hand, increasing the number of iterations provides the model with more opportunities to adjust and refine its parameters, thereby enhancing its ability to converge to the optimal solution over time.

(d)

In the context of random forest models, increasing the number of trees contributes to a more robust model by leveraging the averaging effect across multiple individual trees. This process effectively reduces the model's variance and enhances its overall performance. However, there is a point of decreasing returns where adding more trees does not significantly improve the model's performance, as the variance has already been minimized to an optimal level.

The maximum depth of a decision tree within a random forest indicates the complexity that each tree can achieve. A higher maximum depth allows the model to capture more complex patterns in the data, potentially improving its ability to make accurate predictions. Nevertheless, setting the maximum depth too high can lead to overfitting, where the model becomes too fit to the training data, reducing its performance on unseen data.

In practice, I searched several hyperparameters combinations, observed their changing trends, and selected the best performing set.

(e)

Linear models are the preferred choice when dealing with relatively straightforward data or relationships due to their simplicity, ease of interpretation, and computational efficiency. However, when the data exhibits complexity or nonlinear patterns, linear models may struggle to capture the underlying relationships accurately. In such scenarios, nonlinear models like decision trees work well because they can detect complicated relationships and patterns in the data. Nevertheless, decision trees carry the risk of overfitting, which can be mitigated by employing methods like Random Forests. Random Forests employ a bagging approach, combining multiple decision trees to improve overall performance, but at the cost of increased model complexity.

More complex models can usually perform better than basic models when trained properly, as they can handle more complex data patterns. However, this increased complexity also makes the models' inner workings more challenging for humans to comprehend, resulting in reduced interpretability of the models' decision-making processes.

Reference: chatgpt